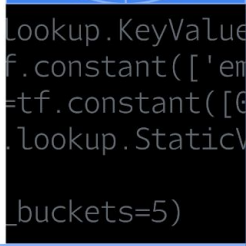
Google Developer Groups

CyberSec Bootcamp Lecture – 0x04



Google Developer Groups

MEET YOUR GUIDES

Zishan Ansari

(Mentor)

CyberSec - GDG

LinkedIn

GitHub

Google Developer Groups

MEET YOUR GUIDES :

Rohit Choudhary

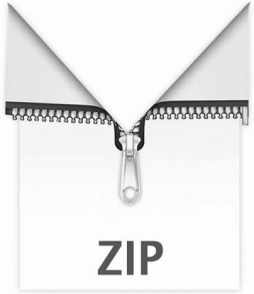
(Instructor)

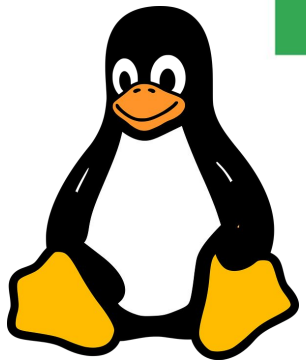
CyberSec - GDG

LinkedIn

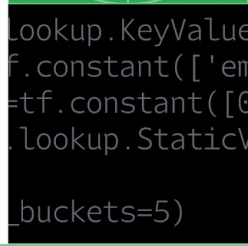
GitHub

Google Developer Groups





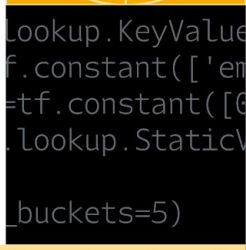
Compressing & Archiving



Google Developer Groups

Why Compress and Archive?

- ❏ Simplifies transferring multiple files.
- ❏ Reduces storage requirements.
- ❏ Common in hacking for downloading/installing software and sharing scripts.
- ❏ Inspired by the Windows `.zip` format.



Google Developer Groups

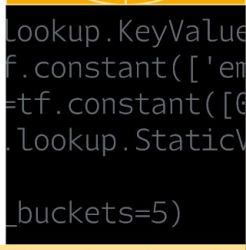
What is Compression?

Process of making data smaller for storage or transfer.

Types:

- **Lossy Compression:** Efficient but loses data integrity (e.g., .jpg, .mp3).
- **Lossless Compression:** Retains data integrity, essential for scripts and software.

Trade-off: Lossless is less efficient than lossy.



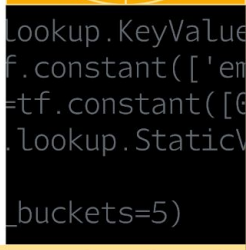
Google Developer Groups

Archiving with tar

Combines multiple files into one archive file (tarball).
Syntax: `tar -cvf <archive_name>.tar <files>`
Options:

- `c` - Create archive.
- `v` - Verbose (optional).
- `f` - Specify filename.

Example: `tar -cvf HackersArise.tar file1 file2 file3`.



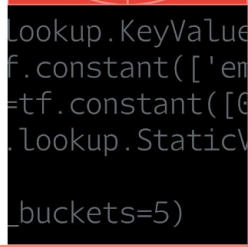
Google Developer Groups

Managing tar Files

View Files: `tar -tvf <archive_name>.tar`.
Extract Files:

- Verbose: `tar -xvf <archive_name>.tar`.
- Silent: `tar -xf <archive_name>.tar`.

Overwrites existing files during extraction.



Google Developer Groups

Compressing tar Files

gzip: Moderate compression and speed.

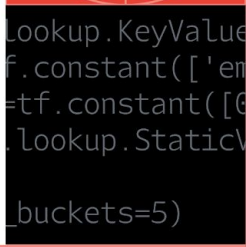
- Syntax: `gzip <file>`.
- File Extension: `.tar.gz` or `.tgz`.

bzip2: Higher compression ratio but slower.

- Syntax: `bzip2 <file>`.
- File Extension: `.tar.bz2`.

compress: Fastest but least effective.

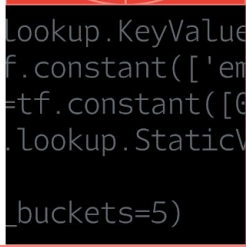
- Syntax: `compress <file>`.
- File Extension: `.tar.Z`.



Google Developer Groups

Extracting Compressed Files

- **gunzip:** Decompress `.gz` files.
- **bunzip2:** Decompress `.bz2` files.
- **uncompress:** Decompress `.Z` files.
- Example:
  - `gunzip <file.gz>` restores the original `.tar` file.



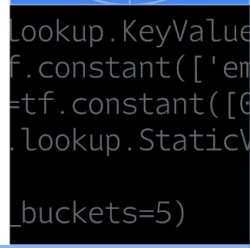
Google Developer Groups

The dd Command

Creates physical copies of storage devices.
Syntax: `dd if=<input> of=<output>`.
Common options:

- `bs`: Block size (default 512 bytes).
- `conv=noerror`: Continue despite errors.

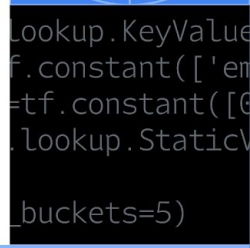
Example: `dd if=/dev/sdb of=/root/flashcopy bs=4096 conv=noerror`.



Google Developer Groups

Why Use These Tools?

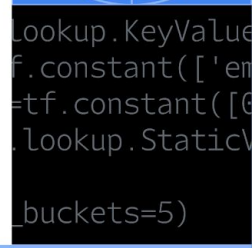
Hackers: Share and store tools/scripts efficiently.
Forensics: Copy drives with deleted files for recovery.
Developers: Bundle and distribute software



Google Developer Groups

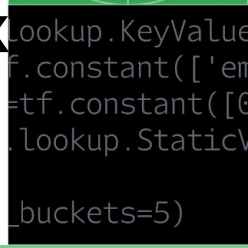
Practice Time!

1. Create 3 scripts and name them: `Linux4Hackers1`, `Linux4Hackers2`, `Linux4Hackers3`.
2. Create a tarball: `tar -cvf L4H.tar <files>`.
3. Compress with gzip, bzip2, and compress. Compare file sizes.
4. Decompress each and ensure data integrity.
5. Use dd to copy a flash drive.



Google Developer Groups

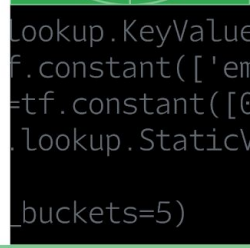
Filesystem & Storage Device Management in Linux



Google Developer Groups

What We Will Cover

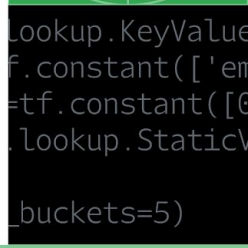
- ❑ Linux filesystem structure overview
- ❑ Device representation in Linux (`/dev`)
- ❑ Mounting and unmounting drives
- ❑ Monitoring and managing storage devices



Google Developer Groups

Linux Filesystem Overview

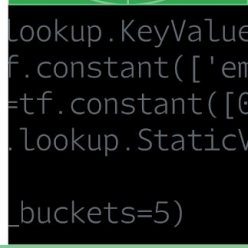
- Hierarchical structure with / (root) at the top
- No drive letters like Windows (C:, D:)
- Unified file tree for all devices



Google Developer Groups

Understanding `/dev` Directory

- Represents all attached devices as files
- Example: `sda`, `sdb` for hard drives
- Character (`c`) and Block (`b`) devices



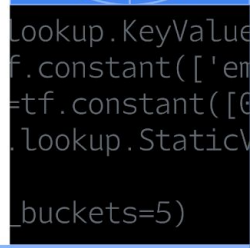
Google Developer Groups

Linux Device Naming

- SATA/IDE drives: `sda`, `sdb`, etc.
- Partitions: `sda1`, `sda2`, etc.
- Incremental naming for multiple devices

Table: Device and Partition Examples

- `sda`: First drive
- `sdb`: Second drive
- `sda1`: First partition on first drive

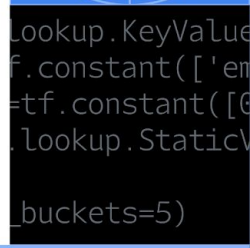


Google Developer Groups

Listing Partitions

- Use `fdisk -l` command
- Displays partitions, sizes, and types

Example: Show a simplified output of `fdisk -l`

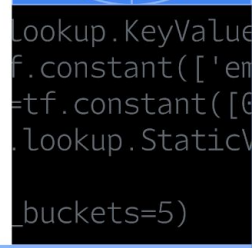


Google Developer Groups

Filesystem Types in Linux

- Linux: `ext2`, `ext3`, `ext4`
- Windows: `NTFS`, `FAT32`
- Compatibility considerations for external drives

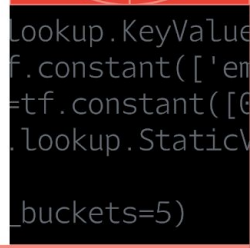
Optional: Comparison table of file systems



Google Developer Groups

Mounting Storage Devices

- Attach storage to the filesystem
- Manual Mount Command: `mount /dev/sdb1 /mnt`
- Auto-mount directories: `/media`, `/mnt`
-

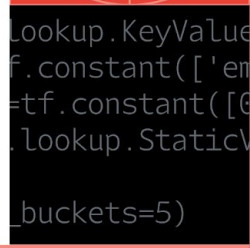


Google Developer Groups

Unmounting Storage Devices

- Command: `umount /dev/sdb1`
- Avoiding data loss by unmounting before removal

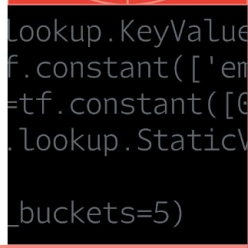
- **Tip**: Mention that `umount` fails if the device is busy.



Google Developer Groups

Monitoring Filesystem and Storage

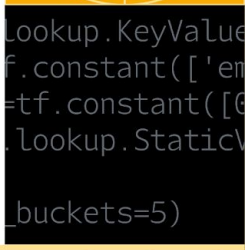
- `df` command: Disk space usage
- `lsblk` command: List block devices
- Checking and fixing errors



Google Developer Groups

Practical Applications for Hackers

- Mounting external drives for tools and data
- Understanding the target filesystem structure
- Recognizing file systems and partitions



Google Developer Groups

Summary

- Linux represents devices as files in `/dev`
- Drives and partitions use logical labels
- Mounting and unmounting are critical for storage management
- Commands: `fdisk`, `mount`, `umount`, `df`, `lsblk`
-